Orca: A Program for Mining Distance-Based Outliers

Mining Distance-Based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule

Introduction

Detecting outliers or anomalies is an important task with many practical applications. Fast algorithms are needed for large databases.

Background

We developed Orca, a program for mining outliers in large multivariate data sets. An outlier is an example that is substantially different from the examples in the remainder of the data. An outlier may have values for an attribute that are unusually large or small, or it may have an unusual combination of values that are rarely seen together.

Orca mines distance-based outliers. That is, Orca uses the distance from a given example to its nearest neighbors to determine its unusualness. The intuition is that if there are other examples that are close to the candidate in the feature space, then the example is probably not an outlier. If the nearest examples are substantially different, then the example is likely to be an outlier. Probabilistically, one can view distance-based outliers as identifying candidates that lie at points where the nearest neighbor density estimate is small.

The key features of Orca

• Orca has excellent scaling properties on large real data sets. Orca can process 1,000,000 census examples in about 20 minutes on a 1.5 Ghz Pentium 4 computer.

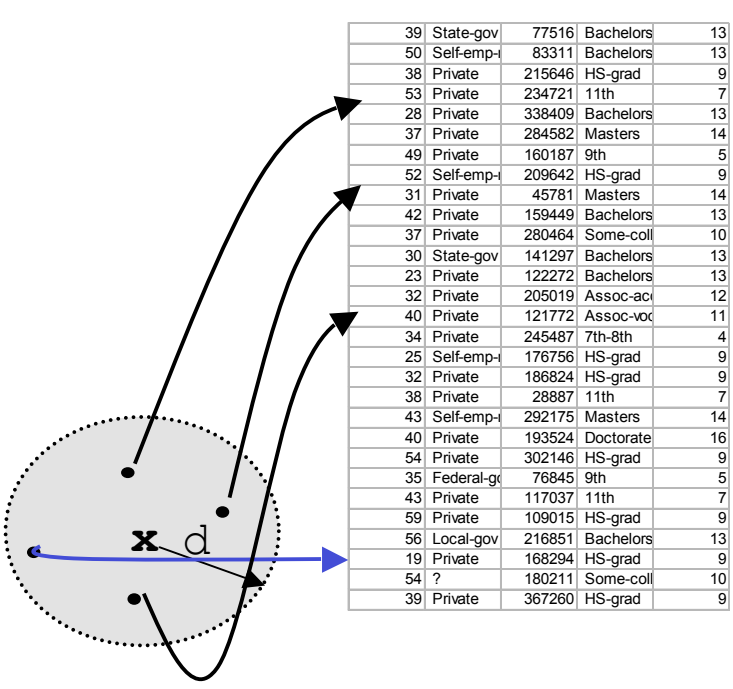
• Orca only requires a limited amount of main memory to run. It does not require loading the entire database into memory. The typical memory footprint is about 3 MB.

• Orca can explain why an example is an outlier. Orca can analyze the features of an example and determine their individual contribution to the unusualness.

• Orca has options to allow users to change the outlier score function and the distance measure

Novel pruning rule

The obvious algorithm for finding distance-based outliers uses nested loops and runs in quadratic time. Orca uses a novel pruning rule to obtain near-linear-time performance, allowing it to scale to very large datasets. The key idea is that while performing the sequential scan for an example's nearest neighbors, the algorithm keeps track of the closest neighbors found so far and prunes examples once the neighbors found so far indicate that the example cannot be a top outlier.



39	State-gov	77516	Bachelors	13
50	Self-emp-i	83311	Bachelors	13
38	Private	215646	HS-grad	9
53	Private	234721	11th	7
28	Private	338409	Bachelors	13
37	Private	284582	Masters	14
49	Private	160187	9th	5
52	Self-emp-i	209642	HS-grad	9
31	Private	45781	Masters	14
42	Private	159449	Bachelors	13
37	Private	280464	Some-coll	10
30	State-gov	141297	Bachelors	13
23	Private	122272	Bachelors	13
32	Private	205019	Assoc-ac	12
40	Private	121772	Assoc-voc	11
34	Private	245487	7th-8th	4
25	Self-emp-i	176756	HS-grad	9
32	Private	186824	HS-grad	9
38	Private	28887	11th	7
43	Self-emp-i	292175	Masters	14
40	Private	193524	Doctorate	16
54	Private	302146	HS-grad	9
35	Federal-g	76845	9th	5
43	Private	117037	11th	7
59	Private	109015	HS-grad	9
56	Local-gov	216851	Bachelors	13
19	Private	168294	HS-grad	9
54	?	180211	Some-coll	10
39	Private	367260	HS-grad	9

In the example above, outliers are based on distance to the 3rd nearest neighbor (k=3), and d is the distance to the 3rd nearest neighbor of the weakest top outlier found so far. When searching for the nearest neighbors of example x, the sequential scan can stop after three neighbors are found within distance d. The example with the blue arrow does not need to be considered; the existence of three examples within distance d is sufficient to prove that x is not a top outlier.

Orca for Liquid Fueled Rocket Engine Fault Detection

Motivation

The ability to detect anomalies in sensor data from a complex engineered system such as a spacecraft is important for at least three reasons. First, detecting anomalies in near-real-time during flight can be helpful in making crucial decisions such as the decision of whether to abort the launch of a spacecraft prior to reaching the intended altitude. Second, for a reusable spacecraft such as the Space Shuttle, detecting anomalies in recorded sensor data after a flight can help to determine that maintenance is or is not needed before the next flight. Third, the detection of recurring anomalies in historical data covering a series of flights can produce engineering knowledge that can lead to design improvements.

Current Approach

Currently, large numbers of human experts watch data in near-real time, aided by limit checks.

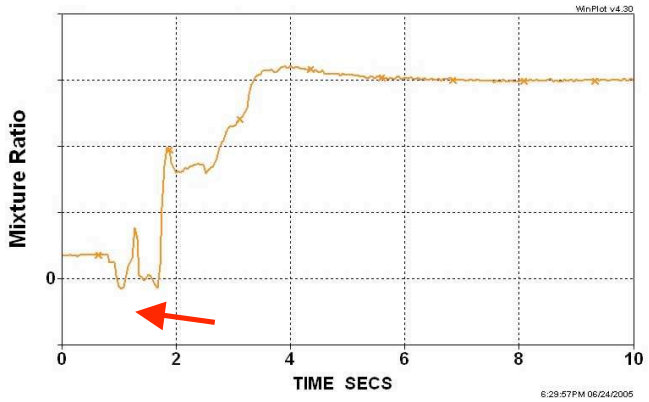
Problems with current approach

- Very labor intensive
- Some faults are too fast for humans to react.
- Humans may not recognize anomalies that involve relationships among large numbers of variables.

Results

We applied Orca to historical data from ground-based test firings of the Space Shuttle Main Engine (SSME). Orca discovered several important anomalies in the SSME data, both previously known and previously unknown, including sensor failures and a failure in the high-pressure fuel turbopump.

SSME Mixture Ratio Anomaly Detected by Orca



In this example, Orca detected a negative mixture ratio. This anomaly is a known artifact of the way in which mixture ratio is calculated (the SSME has no oxidizer flow meter).

Other Applications

Orca has also been applied to:

- Space Shuttle Wing Leading Edge data
- Earth science data
- Aviation security data

Conclusions

In applying outlier detection algorithms to large, real databases, the Orca algorithm finds outliers in many different data sets in near linear time. This efficient scaling allowed us to mine data sets with millions of examples and many features, and to find important anomalies in data from various sources including the Space Shuttle Main Engine. This method is easy to implement and should be the new straw man for research in speeding up distance-based outliers.

Using Ocra:

Instructions, further information, and the Orca software can be found on the following link:

http://www.isle.org/~sbay/software/orca/

Point of Contact:

Dr. Mark Schwabacher
NASA Ames Research Center
Moffett Field, CA
Telephone: 650-604-2409
E-Mail: mark.a.schwabacher@nasa.gov
Web: http://ti.arc.nasa.gov/people/schwabac/

Group Web Page:

http://dataminng.arc.nasa.gov